

BUNDESREPUBLIK DEUTSCHLANDEP 00 / 09741
10 / 089907 4

REC'D 01 DEC 2000

WIPO PCT

**Prioritätsbescheinigung über die Einreichung
einer Patentanmeldung****PRIORITY
DOCUMENT**SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH RULE 17.1(a) OR (b)**Aktenzeichen:** 199 48 099.0**Anmeldetag:** 6. Oktober 1999**Anmelder/Inhaber:** Infineon Technologies AG, München/DE**Bezeichnung:** Prozessorsystem, insbesondere ein Prozessorsystem
für Kommunikationseinrichtungen**IPC:** G 06 F 9/38**Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der ur-
sprünglichen Unterlagen dieser Patentanmeldung.**München, den 20. Oktober 2000
Deutsches Patent- und Markenamt
Der Präsident
Im Auftrag

Wehner

Beschreibung

Prozessorsystem

- 5 Die vorliegende Erfindung betrifft ein Prozessorsystem, insbesondere ein Prozessorsystem für Kommunikationseinrichtungen.

10 Kommunikationscontroller werden in Kommunikationsendgeräten, Routern oder Gateways verwendet, welche nachfolgend der Einfachheit halber als Hostsystem bezeichnet werden. Mit Hilfe der Kommunikationscontroller wird in der Regel der Empfang und das Senden von Kommunikationsinformationen verwaltet bzw. gesteuert. Zum Senden werden die in Form von
15 digitalen Daten vorliegenden Kommunikationsinformationen aus einem Puffer geholt und anschließend, eventuell verpackt mit weiteren Informationsbits, in einen Sendepuffer geschrieben. Der Inhalt des Sendepuffers wird anschließend an einen Modulator oder Transceiver weitergereicht, welcher die
20 digitalen Sendedaten auf ein Trägersignal aufmoduliert, D/A-wandelt und über einen Kommunikationskanal an einen Empfänger sendet.

Die Verarbeitung von Kommunikationsinformationen erfolgt in Übereinstimmung mit sogenannten Kommunikationsprotokollen, welche in Form von jeweils als "Layer" bezeichneten Schichten aufgebaut sind. In Senderichtung müssen in jedem Layer die Eingangsdaten des jeweiligen Layers mit einem sogenannten Protokollheader verpackt und in den untergeordneten Layer
30 weitergereicht werden. In Empfangsrichtung muß hingegen den jeweiligen Eingangsdaten der Protokollheader entnommen und die Daten an den jeweils übergeordneten Layer weitergereicht werden. Somit sind neben der Auswertung und Ausführung von beispielsweise über einen Hostrechner oder Microcontroller
35 extern zugeführten Steuerbefehlen die Hauptaufgaben eines Kommunikationscontrollers die Extraktion der Headerinformationen aus einem empfangenen Bitstrom, das



Hinzufügen der Headerinformationen zu einem zu sendenden Bitstrom und das Weiterreichen der jeweiligen Daten an einen anderen Layer (d.h. der Datentransfer).

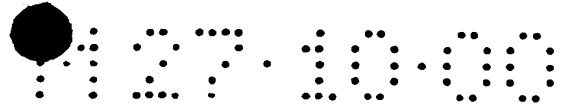
- 5 Hinsichtlich der Architektur von Kommunikationscontrollern sind verschiedene Vorschläge bekannt, welche im wesentlichen durch die jeweils verwendete zentrale Prozessoreinheit (CPU), die Speicherstruktur, die Busstruktur oder den Befehlssatz charakterisiert werden können. Eine typische
- 10 Kommunikationscontroller-Architektur ist beispielsweise in "A 16Mb/s Adapter Chip for the IBM Token-Ring Local Area Network", J.D. Blair et. al., IEEE Journal of Solid-State Circuits, vol. 24, Dezember 1989, beschrieben. Der oben
- 15 beschriebene Datentransfer stellt regelmäßig den Flaschenhals des Kommunikationscontrollers darstellt. Dies hat bei der zuvor erwähnten bekannten Architektur zur Folge, daß das auch als "Task Switch" bezeichnete Umschalten von einem Programmablauf auf einen anderen Programmablauf relativ lange dauert.

20

Aus diesem Grund wurden Kommunikationscontroller vorgeschlagen, deren Architektur die parallele Abarbeitung verschiedener Befehle oder Programmabläufe ermöglicht. So wird beispielsweise in der US-Patentschrift 5,434,976 ein

25 Kommunikationscontroller vorgeschlagen, welcher zwei voneinander unabhängige zentrale Prozessoreinheiten (CPUs) aufweist, wobei jede Prozessoreinheit ihren eigenen Pfad zum Holen oder Auslesen eines auszuführenden Befehls und zum Decodieren und Ausführen des ausgelesenen Befehls besitzt.

- 30 Mit Hilfe der einen Prozessoreinheit wird im wesentlichen die Funktion des sogenannten MAC-Layers ("Medium Access Control") wahrgenommen, während die andere Prozessoreinheit im wesentlichen Hostbefehle ausführt und die mit dem Empfang oder dem Senden von Daten verbundenen
- 35 Pufferspeicherfunktionen verwaltet.



Obwohl mit Hilfe dieser bekannten Architektur eine wirkungsvolle Verarbeitung der Kommunikationsinformationen, insbesondere ein relativ schneller Datentransfer, möglich ist, besitzt diese Architektur den Nachteil, daß zwei
5 getrennte Prozessoreinheiten verwendet werden, welche die benötigte Chipfläche sowie den Leistungsverbrauch erhöhen.

Der vorliegenden Erfindung liegt daher die Aufgabe zugrunde, ein Prozessorsystem vorzuschlagen, welches einerseits einer
10 geringen Chipflächenbedarf und Leistungsverbrauch aufweist und andererseits eine wirkungsvolle und schnelle Ausführung der bei der Verarbeitung von Kommunikationsinformationen erforderlichen Funktionen ermöglicht.

15 Diese Aufgabe wird erfindungsgemäß durch ein Prozessorsystem mit den Merkmalen des Anspruches 1 gelöst. Die Unteransprüche definieren bevorzugte und vorteilhafte Ausführungsformen der vorliegenden Erfindung.

20 Das insbesondere in Form eines Kommunikationscontrollers ausgebildetes erfindungsgemäße Prozessorsystem umfaßt lediglich eine zentrale Prozessoreinheit (CPU) zur Ausführung von in einem Programmspeicher abgelegten Befehlen, wobei die Prozessoreinheit einen einzigen Pfad zum Auslesen eines
25 Befehls aus dem Programmspeicher und zum Decodieren des ausgelesenen Befehls umfaßt. Des weiteren sind mehrere parallel betreibbare Ausführungspfade zum parallelen Ausführen verschiedener Befehle oder Programmabläufe vorgesehen, welche jeweils auf den gemeinsam zum Auslesen und
30 Decodieren eines Befehls genutzten Pfad zugreifen.

Auf diese Weise ist prinzipiell die parallele Abarbeitung unterschiedlicher Programmabläufe möglich, wobei der
Chipflächenbedarf und Leistungsverbrauch durch den gemeinsam
35 genutzten Pfad zum Auslesen und Decodieren der Befehle minimiert werden kann.

27.10.00

Vorteilhaft ist es, wenn lediglich einer der Ausführungspfade die normalen Rechen- und Adressierfunktionen der CPU wahrnimmt, während ein anderer Ausführungspfad nur bestimmte spezielle Funktionen ausführt, so daß dieser Ausführungspfad einfacher implementiert und der Realisierungsaufwand weiter verringert werden kann. Besonders vorteilhaft ist es, wenn der zuletzt genannte Ausführungspfad nur eine besonders häufig benötigte Funktion ausführen kann, wobei dies bei der Verarbeitung von Kommunikationsprotokollen beispielsweise das Verschieben von Datenblöcken, d.h. der Datentransfer, sein kann.

Zum Verschieben, d.h. Laden oder Speichern, von Datenblöcken kann der Befehlssatz des Prozessorsystems vorteilhafterweise derart erweitert werden, daß mit dem entsprechenden "Block Move"-Befehl zugleich ein programmierbarer Offsetwert vorgegeben wird, mit dem der Datenblock beispielsweise in einen Ausgabeport geschrieben oder aus einem Eingabeport ausgelesen wird. Dabei kann als Offsetwert insbesondere ein Bitoffset verwendet werden.

Durch die Verwendung von Datenbussen unterschiedlicher Übertragungsraten können zudem weniger häufig benötigte oder langsamere Funktionen auf dem Datenbus mit der niedrigeren Übertragungsrate ausgeführt werden, während häufiger benötigte oder schnell auszuführende Funktionen, wie beispielsweise insbesondere Datentransfers, über den Datenbus mit der höheren Übertragungsrate ausgeführt werden können. Dieses erfindungsgemäße Ausführungsbeispiel unterscheidet sich somit von den bekannten Lösungen, welche entweder die Verwendung eines einzelnen Datenbus in Kombination mit einer einzigen Prozessoreinheit oder die Verwendung von zwei separaten Datenbussen in Kombination mit zwei separaten Prozessoreinheiten vorschlagen.

Im Falle eines Kommunikationscontrollers können auf diese Weise auf dem mit einer ausreichenden Bandbreite versehenen

: 1 2 7 . 1 0 . 0 0

Datenbus, der die höhere Übertragungsrate aufweist, schnelle Transfers der sogenannten Payloaddaten ausgeführt werden, während die Verarbeitung der Protokoll-Headerdaten über den langsameren Datenbus erfolgt.

5

Weitere Vorteile der vorliegenden Erfindung, welche aus der zuvor beschriebenen Architektur des erfindungsgemäßen Prozessorsystems resultieren, sind u.a. die Verwendung unterschiedlicher Bitbreiten für die einzelnen Datenbusse und einer heterogenen Speicherhierarchie für die Ports, Register und den RAM-Datenspeicher des Prozessorsystems sowie die Möglichkeit der Implementierung unterschiedlicher Energiesparmodi für jeden Ausführungspfad und jeden Datenbus.

10

Die vorliegende Erfindung wird nachfolgend unter Bezugnahme auf die beigefügte Zeichnung anhand bevorzugter Ausführungsbeispiele näher erläutert.

15

Fig. 1 zeigt ein vereinfachtes Blockschaltbild einer zentralen Prozessoreinheit oder Zentraleinheit (CPU) gemäß einem Ausführungsbeispiel der vorliegenden Erfindung,

20

Fig. 2 zeigt die Anbindung der in Fig. 1 gezeigten Prozessoreinheit an ein erfindungsgemäßes Prozessorsystem, und

25

Fig. 3A-3C zeigen vergrößerte Darstellungen von in Fig. 2 gezeigten Ein-/Ausgabeports und Registern.

Wie in Fig. 1 gezeigt ist, umfaßt die als Kommunikationscontroller für beispielsweise ein Kommunikationsendgerät dienende Prozessoreinheit 1 (CPU) einen Pfad, um aus einem Programmspeicher 8 des Prozessorsystems einen Befehl auszulesen und zu decodieren, was in Fig. 1 in Form von Funktionsblöcken 2 und 3 dargestellt ist.

30

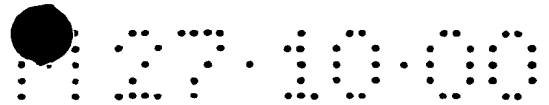
35

1.27.10.00

Um die bei der Verarbeitung von Kommunikationsdaten durchzuführenden Datentransfers sowie die Protokollverarbeitung zu optimieren, ist eine parallele Befehlsausführung vorgesehen, wobei die Parallelität erst nach den Funktionsblöcken 2 und 3 gegeben und durch mehrere parallel betreibbare Ausführungseinheiten 5, 7 gebildet ist. Insbesondere sind bei dem in Fig. 1 gezeigten Ausführungsbeispiel zwei derartige parallele Ausführungseinheiten 5, 7 zur unabhängigen Abarbeitung verschiedener Befehle oder Programmabläufe vorgesehen, wobei jeder Ausführungseinheit nach der Decodierstufe 3 ein Pufferspeicher oder Register 4 bzw. 6 zugeordnet ist. Diese Pufferspeicher 4, 6 dienen jeweils zum Speichern von Informationen, welche die nachfolgende Ausführungseinheit 5 bzw. 7 zur Befehlsausführung benötigt.

Gemäß der in Fig. 1 gezeigten Architektur wird somit lediglich ein einziger Pfad 2, 3 zum Holen bzw. Auslesen und Decodieren eines Befehls verwendet, der von sämtlichen parallel betreibbaren Ausführungseinheiten 5, 7 gemeinsam genutzt wird, wodurch die Implementierung und Programmierung vereinfacht wird.

Die Ausführungseinheiten 5 und 7 sind vorteilhafterweise unterschiedlich aufgebaut bzw. strukturiert. Eine dieser Ausführungseinheiten sollte zur Reduzierung des Realisierungsaufwands so einfach wie möglich aufgebaut sein, um lediglich eine begrenzte Anzahl von Funktionen oder Befehle ausführen zu können, während mindestens eine andere Ausführungseinheit so allgemein wie gewünscht gehalten werden kann und insbesondere alle möglichen Funktionen bzw. Befehle ausführen können sollte. Bei dem dargestellten Ausführungsbeispiel ist die Ausführungseinheit 7 sogar so weit vereinfacht, daß sie nur zur Durchführung eines bestimmten Befehls, vorzugsweise des Befehls zum Verschieben eines Datenblocks, ausgestaltet ist, während die Ausführungseinheit 5 den gesamten Befehlssatz des



Prozessorsystems ausführen kann. Auf diese Weise kann der Aufbau des Puffers 6 und der Ausführungseinheit 7 gegenüber dem Aufbau des Puffers 4 und der Ausführungseinheit 5 vereinfacht werden.

5

Wie bereits erwähnt worden ist, dienen die Puffer 4 und 6 jeweils zum Zwischenspeichern von Informationen, welche für die Durchführung des in dem jeweiligen Ausführungspfad auszuführenden Befehls erforderlich sind. Bei der Ausführung von Befehlen, welche mehrere Zyklen dauern (wie z.B. bei einem Datenblock-Verschiebepfehl), werden die jeweils benötigten Informationen in den Puffern 4 bzw. 6 gespeichert, um auf diese Weise den Pfad mit den Funktionsblöcken 2 und 3 für die parallele Durchführung eines anderen Befehls freizumachen. Dient die Ausführungseinheit 7 lediglich zur Ausführung des Befehls zur Verschiebung eines Datenblocks, muß in den entsprechenden Puffer 6 lediglich ein Bit zur Unterscheidung zwischen einem Lade- und einem Speicherbefehl, die Adresse der Datenquelle bzw. der Datensenke, die Anzahl der zu verschiebenden Datenwörter (z.B. Bytes) sowie gegebenenfalls eine nachfolgend noch näher erläuterte Offsetinformation (z.B. eine Bitoffsetinformation) gespeichert werden.

Ist die Ausführungseinheit 7 zur Durchführung eines Datentransfers aktiv, kann der Ausführungspfad der Ausführungseinheit 5 parallel mit einem weiteren Programmablauf ("Task") belegt werden. Jedem Programmablauf sind seine eigenen Zustandsvariablen zugeordnet, welche den auch als "Kontext" bezeichneten Zustandsraum des jeweiligen Programmablaufs bilden. Für jeden aktiven Programmablauf sollte eine eigene Hardware, wie z.B. Programmzähler und Register zum Speichern der jeweiligen Zustandsvariablen (Zero-Bits, Carry-Bits, Adress Pointer etc.), vorgesehen sein, so daß beim Umschalten von einem Programmablauf auf einen anderen die jeweils vorhandene Hardware mit den

entsprechenden Zustandsvariablen umgeschaltet wird und eine Kontext-Umschaltung ohne Zyklenverlust möglich ist.

Für den sogenannten Multitasking- oder Parallelbetrieb wird ein Befehlssatz benötigt, der u.a. Befehle zum Starten und Stoppen eines Programmablaufs, zum Setzen einer programmablaufspezifischen Priorität oder zum Warten auf ein Signal zur Programmablauf-Synchronisation umfaßt. Für den Fall, daß die Ausführungseinheit 5 keinen Befehl mehr auszuführen hat, soll vorgesehen sein, daß der gesamte Pfad vom Funktionsblock 2 bis zum Funktionsblock 5 sowie die entsprechenden Register für die Zustandsvariablen abgeschaltet werden können, was ebenfalls über einen expliziten Befehl zur Taktabschaltung geschehen kann.

Da jede der beiden Ausführungseinheiten 5, 7 unterschiedlichen Programmabläufen zugeordnet sind, wird die Kommunikation zwischen der ALU ("Algorithmic Logical Unit"), den Registern, RAM-Speichern und Datenbusses etc. des Prozessorsystem vereinfacht. Da der Datentransfer eines Programmablaufs in der Ausführungseinheit 7 parallel zu einem anderen Programmablauf, welcher von der Ausführungseinheit 5 abgearbeitet wird, ablaufen kann, kann im Vergleich zu einem DMA-basierten Prozessorsystem ("Direct Memory Access") auch dann eine bessere Kontrolle innerhalb der Software erzielt werden, wenn der Datenverschiebepfehl unterbrechen werden müßte. Zudem ist keine Verwaltung von DMA-Controllern erforderlich, was zu einer Reduzierung des Programmcodes und des Energiebedarfs beiträgt. Zudem werden die Hardwarekosten verringert, da die Register oder Puffer 4, 6 mehrmals verwendet werden können und keine zusätzlichen Register für die DMA-Controller erforderlich sind.

Wie bereits erwähnt worden ist, dient die Ausführungseinheit 7 bevorzugt lediglich zur Durchführung eines Datenblock-Verschiebepfehls, während die Ausführungseinheit 5 so allgemein wie möglich ausgestaltet sein kann, um alle

: 1 2 7 . 1 0 . 0 0

möglichen Funktionen oder Befehle ausführen zu können. Bei der Verarbeitung von Kommunikationsdaten kann die Ausführungseinheit 7 somit zur Durchführung von Datentransfers verwendet werden, während die Ausführungseinheit 5 parallel dazu die restliche Protokollverarbeitung übernimmt. Da die Datentransfers im Vergleich zu der Protokollverarbeitung relativ schnell ablaufen müssen, ist es vorteilhaft, die Ausführungseinheit 7 an einen Datenbus mit relativ hoher Übertragungsrate anzuschließen. Dies soll nachfolgend näher anhand Fig. 2 erläutert werden.

Die zentrale Prozessoreinheit 1 ist innerhalb des Prozessorsystem mit zwei Datenbussen 9 und 10 verbunden, wobei der Datenbus 9 eine höhere Übertragungsrate aufweist als der Datenbus 10. Die Prozessoreinheit 1 kann mit ihrer Ausführungseinheit 7 lediglich auf den Datenbus 9 zugreifen, während die allgemein aufgebaute Ausführungseinheit 5 beide Datenbusse 9, 10 nutzen kann. Zur Realisierung der schnellen Datentransfers, insbesondere zur Verschiebung der sogenannten Payloaddaten von Kommunikationsprotokollen, welche die eigentlichen Kommunikationsinformationen aufweisen, sind Ein- und Ausgabeports 11 sowie Puffer oder Register 12 an den schnellen Datenbus 9 angeschlossen. Die Ports 11 sind als aufwendigere Spezialregister realisiert und dienen als Schnittstelle zu einem auch als Modulator bezeichneten Transceiver 14 des entsprechenden Kommunikationsendgeräts, über den Daten gesendet bzw. empfangen werden. Ein Interrupt Controller 16 wertet insbesondere von den Ports 11 generierte Interruptsignale aus und steuert davon abhängig in Übereinstimmung mit einer vorgegebenen Interruptbehandlung die zentrale Prozessoreinheit 1 an. Die Puffer 12 bilden die Schnittstelle zu dem jeweiligen Hostsystem 15 (d.h. dem jeweiligen Kommunikationsendgerät, Router oder Gateway), welches den zu übertragenden bzw. empfangenden Bitstrom in die Puffer 12 schreibt bzw. ausliest. Die beispielsweise als RAM-Speicher realisierten Puffer 12 können über von der



Prozessoreinheit 1 konfigurierbare DMA-Controller gesteuert werden, so daß die Datentransfers zwischen den Puffern 12 und dem Hostsystem 15 keine nennenswerte Belastung für die Prozessoreinheit 1 darstellen. Der langsamere Datenbus 10 ist
5 hingegen mit dem eigentlichen Datenspeicher 13 des Prozessorsystems verbunden, der in Form eines RAM-Massenspeichers ausgestaltet ist und vornehmlich zum Zwischenspeichern von Steuerinformationen und Headerinformationen eines Kommunikationsprotokolls dient.

10

Auf diese Weise sind die bei der Verarbeitung von Kommunikationsprotokollen für den Transfer der sogenannten Payloaddaten vorgesehenen Komponenten 11, 12 von dem für die Verarbeitung der Protokoll-Headerdaten vorgesehenen Abschnitt
15 getrennt, und für die schnell durchzuführenden Payloaddatentransfers kann die Ausführungseinheit 7 sowie der schnelle Datenbus 9 genutzt werden, während über den langsameren Datenbus 10 (und die Ausführungseinheit 5) dazu parallel die Verarbeitung der Headerdaten des jeweiligen
20 Kommunikationsprotokolls ablaufen kann.

In Fig. 3A sind die einem Port 11 mit serielllem Dateneingang und parallelem Datenausgang zugeführten Signale dargestellt. Dieser Port 11 empfängt seriell Eingangsdaten D_IN, welche
25 entsprechend einem Taktsignal CLK in den Port 11 geschrieben werden. Zudem empfängt der Port 11 eine 3 Bit-Steuerinformation BP, welche die Bitposition des einzuschreibenden Daten bezeichnet, sowie eine weitere 3 Bit-Steuerinformation BW, welche die Breite oder Länge des
30 einzuschreibenden Bitfeldes bezeichnet. Auf diese Weise wird in den Port 11 ein Datenblock der Länge BW mit einem auf die erste Bitposition des Ports 11 bezogenen Bitoffset der Länge BP geschrieben. Als weiteres Steuersignal wird ein
35 Resetsignal RESET dem Port 11 zugeführt. Ausgangsseitig werden die Daten mit n Bits parallel ausgelesen, wobei über ein weiteres Steuersignal D_READ das Lesen der Daten freigegeben wird. Darüber hinaus ist ein Interruptsignal

127.10.00

D_READY vorgesehen, welches einen Interrupt erzeugt, wenn in dem Port 11 Daten gespeichert sind.

5 In Fig. 3B sind die einem Port 11 mit parallelem Dateneingang und seriellem Datenausgang angelegten Signale dargestellt, wobei sich dieser Port von dem in Fig. 3A gezeigten Port lediglich dadurch unterscheidet, daß eingangsseitig n-Bits parallel eingelesen und die Ausgangsdaten seriell ausgegeben werden.

10

Schließlich sind in Fig. 3C auch die an einen Puffer oder ein Register 12 angelegten Signale dargestellt, wobei in die Puffer 12 vorzugsweise Datenwörter in Form von Bytes geschrieben und ausgelesen werden. Zudem wird ein Adreßsignal
15 ADR angelegt, welches jeweils die Adresse des Puffers 12 bezeichnet, auf die zugegriffen werden soll.

1.27.10.00

Patentansprüche

1. Prozessorsystem,

mit einer Prozessoreinheit (1) zur Ausführung von in einem
5 Programmspeicher (8) abgelegten Befehlen,
wobei die Prozessoreinheit (1) Befehlsauslesemittel (2) zum
Auslesen eines Befehls aus dem Programmspeicher (8),
Befehlsdecodiermittel (3) zum Decodieren des Befehls und
Befehlsausführungsmittel (4-7) zum Ausführen des Befehls
10 umfaßt,

d a d u r c h g e k e n n z e i c h n e t,
daß die Befehlsausführungsmittel (4-7) mehrere parallel
betreibbare Ausführungseinheiten (5,7) zum parallelen
Ausführen verschiedener Befehle umfassen, und
15 daß die Befehlsauslesemittel und die Befehlsdecodiermittel
(3) für sämtliche Ausführungseinheiten (5,7) gemeinsam
vorgesehen sind.

2. Prozessorsystem nach Anspruch 1,

20 d a d u r c h g e k e n n z e i c h n e t,
daß jeder Ausführungseinheit (5,7) Zwischenspeichermittel
(4,6) zum Speichern von für die Ausführung des von der
jeweiligen Ausführungseinheit (5,7) auszuführenden Befehls
erforderlichen Informationen zugeordnet sind.

3. Prozessorsystem nach Anspruch 1 oder 2,

d a d u r c h g e k e n n z e i c h n e t,
daß eine erste Ausführungseinheit (5) der
Befehlsausführungsmittel (4-7) zur Ausführung aller möglichen
30 Befehle des Prozessorsystems ausgestaltet ist, und
daß eine zweite Ausführungseinheit (7) der
Befehlsausführungsmittel (4-7) zur Ausführung lediglich
einiger spezieller Befehle ausgestaltet ist.

4. Prozessorsystem nach Anspruch 3,

35 d a d u r c h g e k e n n z e i c h n e t,

4.27.10.00

daß die zweite Ausführungseinheit (7) zur Ausführung einiger häufig verwendeter Befehle ausgestaltet ist.

5. Prozessorsystem nach Anspruch 3 oder 4,
5 d a d u r c h g e k e n n z e i c h n e t,
daß die zweite Ausführungseinheit (7) zur Ausführung lediglich eines speziellen Befehls ausgestaltet ist.

6. Prozessorsystem nach Anspruch 5,
10 d a d u r c h g e k e n n z e i c h n e t,
daß die zweite Ausführungseinheit (7) lediglich zur Ausführung eines Befehls zur Verschiebung eines Datenblocks ausgestaltet ist.

15 7. Prozessorsystem nach Anspruch 6 und Anspruch 2,
d a d u r c h g e k e n n z e i c h n e t,
daß die in den der zweiten Ausführungseinheit (7) zugeordneten Zwischenspeichermitteln (6) gespeicherten Informationen eine Speicher- bzw. Ladeadresse des zu
20 speichernden bzw. zu ladenden Datenblocks, die Anzahl der zu verschiebenden Datenwörter des Datenblocks, einen Offsetwert, mit dem der Datenblock gespeichert bzw. gelesen werden soll und/oder Steuerinformationen, welche angeben, ob es sich bei dem auszuführenden Befehl um einen Speicher- oder Lesebefehl
5 handelt, umfassen.

8. Prozessorsystem nach einem der Ansprüche 3-7,
d a d u r c h g e k e n n z e i c h n e t,
daß die Prozessoreinheit (1) derart ausgestaltet ist, daß der
30 von den Befehlsauslesemitteln (2) über die Befehlsdecodiermittel (3) zu der ersten Ausführungseinheit (5) führende Pfad vorübergehend deaktiviert wird, falls von der ersten Ausführungseinheit (5) augenblicklich kein Befehl auszuführen ist.

35

9. Prozessorsystem nach einem der vorhergehenden Ansprüche,
d a d u r c h g e k e n n z e i c h n e t,



daß eine Ausführungseinheit (5) der Befehlsausführungsmittel (4-7) mit einem ersten Datenbus (10) und eine zweite Ausführungseinheit (5) mit einem zweiten Datenbus (9) verbunden ist, wobei die Übertragungsgeschwindigkeit des ersten Datenbus (10) niedriger als die Übertragungsgeschwindigkeit des zweiten Datenbus (9) ist.

10. Prozessorsystem nach Anspruch 9, dadurch gekennzeichnet, daß das Prozessorsystem (8) zur Verarbeitung von Telekommunikationsprotokollen vorgesehen ist, und daß der erste Datenbus (10) zur Verarbeitung von Headerdaten der Telekommunikationsprotokolle vorgesehen ist, während der zweite Datenbus (11) für einen schnellen Transfer von Payloaddaten vorgesehen ist.

11. Prozessorsystem nach Anspruch 9 oder 10, dadurch gekennzeichnet, daß an den ersten Datenbus (10) ein Datenspeicher (13) des Prozessorsystems angeschlossen ist, und daß an den zweiten Datenbus (9) mindestens ein Ein- und/oder Ausgabeport (11) und/oder mindestens ein Register oder Puffer (12) angeschlossen ist.

12. Prozessorsystem nach Anspruch 10 und 11, dadurch gekennzeichnet, daß der an den zweiten Datenbus (9) angeschlossene Ein- und/oder Ausgabeport mit einer Sende- und/oder Empfangseinheit (14) eines Kommunikationsendgeräts verbunden ist, und daß das an den zweiten Datenbus (9) angeschlossene Register oder Puffer (12) zum Zwischenspeichern eines von dem Kommunikationsendgerät zu sendenden bzw. zu empfangenden Bitstroms vorgesehen ist.

13. Prozessorsystem nach einem der Ansprüche 9-12 und einem der Ansprüche 3-8,

N 27.10.00

15

dadurch gekennzeichnet,
daß die mit dem ersten Datenbus (10) verbundene
Ausführungseinheit der Befehlsausführungsmittel (4-7) der
ersten Ausführungseinheit (5) und die mit dem zweiten
5 Datenbus (9) verbundene Ausführungseinheit der zweiten
Ausführungseinheit (7) entspricht.

14. Prozessorsystem nach Anspruch 13,
dadurch gekennzeichnet,
10 daß die erste Ausführungseinheit (5) zudem mit dem zweiten
Datenbus (9) verbunden ist, um auch auf den zweiten Datenbus
(9) zugreifen zu können, während die zweite
Ausführungseinheit (7) nur mit dem zweiten Datenbus (9)
verbunden ist.

15

Zusammenfassung

Prozessorsystem

- 5 Ein insbesondere in Form eines Kommunikationscontrollers
ausgebildetes Prozessorsystem umfaßt eine zentrale
Prozessoreinheit (1) zur Ausführung von in einem
Programmspeicher (8) abgelegten Befehlen, wobei die
Prozessoreinheit (1) lediglich einen Pfad (2,3) zum Auslesen
10 eines Befehls aus dem Programmspeicher (8) und zum Decodieren
des Befehls umfaßt. Des weiteren sind mehrere parallel
betreibbare Ausführungspfade (4,5; 6,7) zum parallelen
Ausführen verschiedener Programmabläufe vorgesehen, welche
jeweils auf den zum Auslesen und Decodieren eines Befehls
15 gemeinsam genutzten Pfad (2,3) zugreifen.

(Fig. 1)

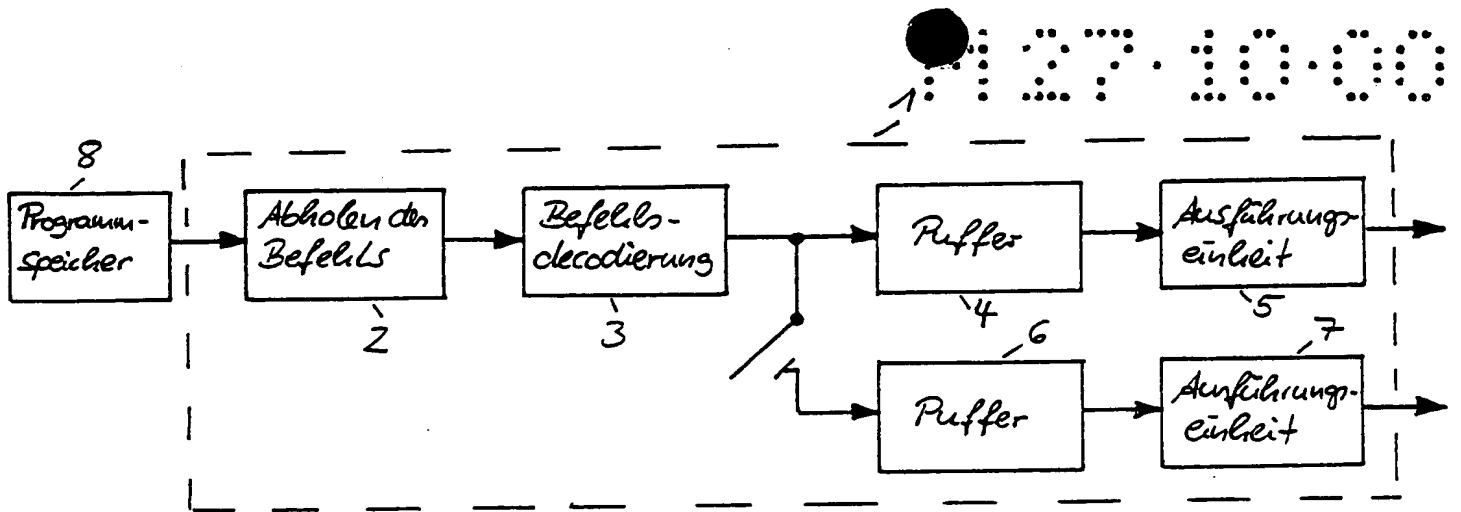


FIG. 1

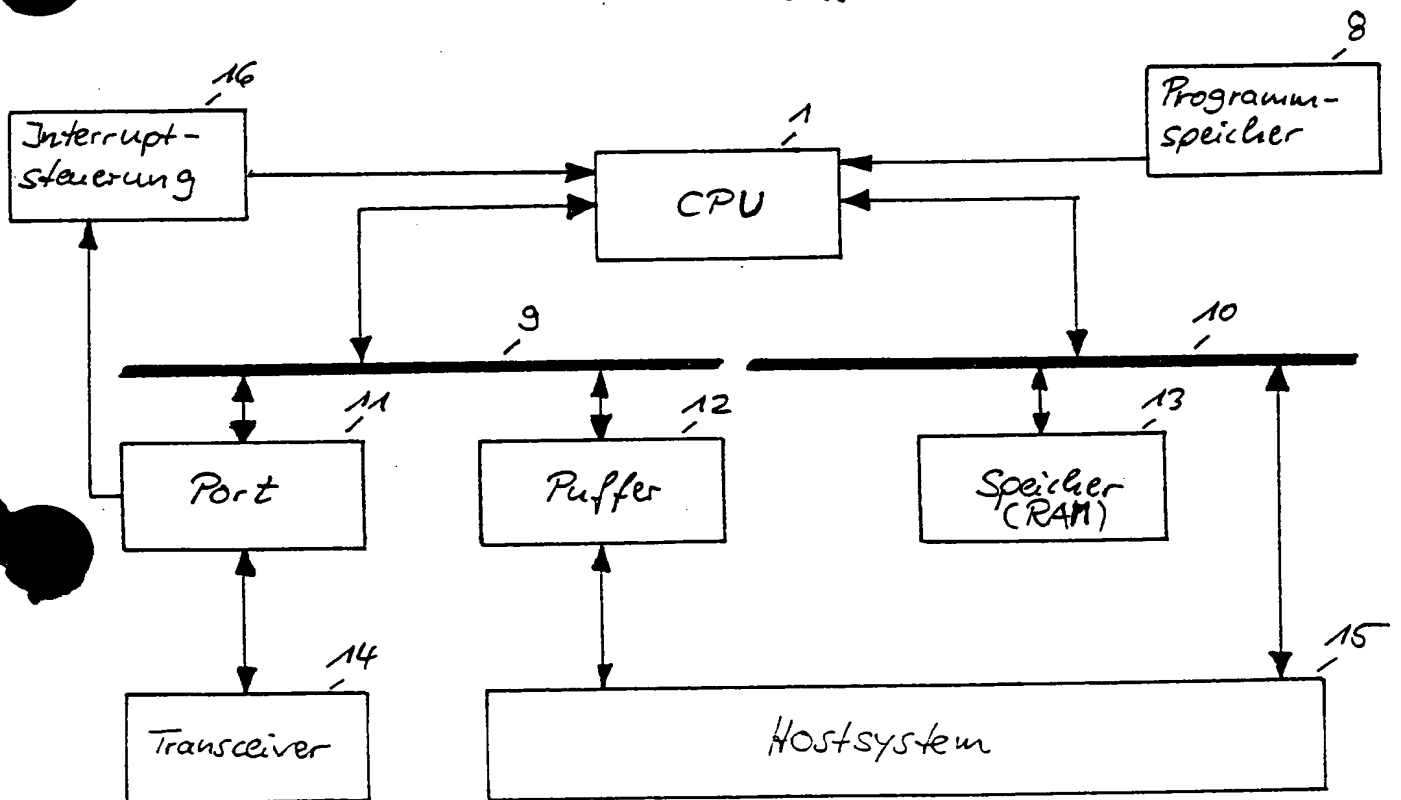
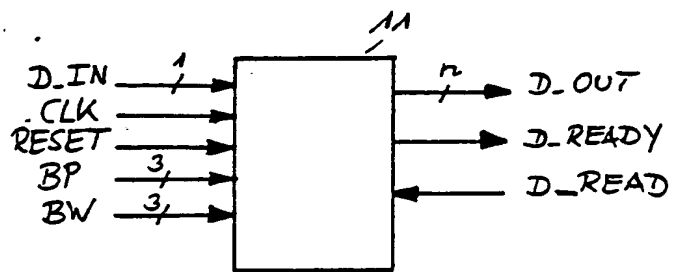
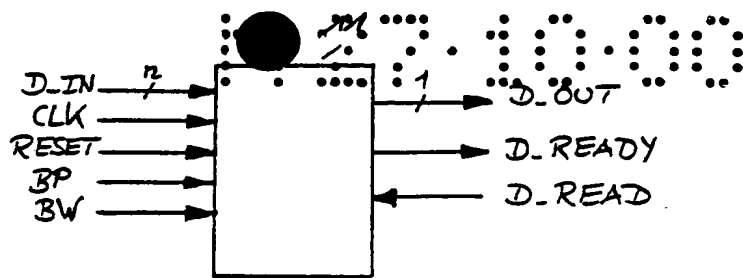


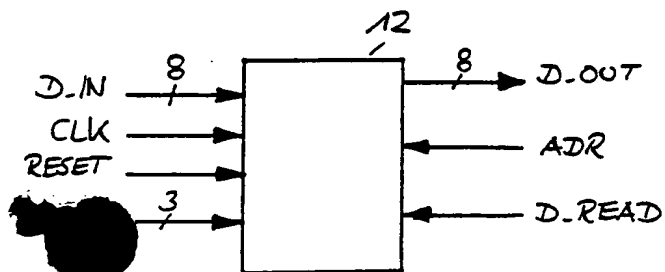
FIG. 2



A)



B)



C)

FIG. 3

THIS PAGE BLANK (USPTO)